

REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 1-11, 13-15, and 17-28 are pending in the application. Claims 34-39 have been canceled without prejudice or disclaimer of the subject matter therein. Claims 1, 13, and 22 are independent claims and each of these claims have been amended hereby.

The Office Action of March 15, 2006 has been carefully considered by the Applicant. Claims 1-11, 13-15, 17-28 and 34-39 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,130,911 to Parker et al. (hereinafter referred to as the Parker reference) in view of US Patent No. 6,415,396 to Singh et al. (hereinafter referred to as the Singh reference).

In overview, by the present Amendment, the rejections have been traversed in view of the following remarks. The Applicant respectfully requests reconsideration and allowance of the subject application. This Amendment is believed to be fully responsive to all issues raised in the Office Action of March 15, 2006.

Claim Rejection Under 35 USC §103(a)

Claims 1-11, 13-15, 17-28 and 34-39 stand rejected under 35 U.S.C. §103(a) as being unpatentable over the Parker reference in view of the Singh reference. For at least some of the reasons that follow, Applicant respectfully disagrees that the subject matter of the above claims is obvious given the above cited references.

1 *Review of the Cited References*

2 The Parker reference describes automatic testing using a script. In
3 particular, the Parker reference describes a way in which the script could be
4 written so that the script could be used to test the application on multiple
5 platforms, such as OPENLOOK, MOTIF, MACINTOSH, and MICROSOFT
6 WINDOWS environments.

7 Table 2, Columns 2 and 9 of the Parker reference shows several example
8 script entries that may be used during a testing process. The Table 2 shows that
9 entries must first be entered into the script before they can be used to test a GUI.
10 Parker explicitly states that the "test executive passes the GUI specific command
11 to the test driver. The test driver then performs the actual action o the GUI object
12 specified in the test script command. (See column 8, lines 20-24.)

13 The Singh reference, as described in the Summary, is directed at an
14 apparatus and method for generating and maintaining a regression test case set
15 directly and automatically from requirement models. The Summary further
16 defines a regression test case as a set of test cases that are used by the user to
17 verify functionality in view of modifications or additions to the system. The Singh
18 reference describes the use of a graph representation for automatically generating
19 an optimum number of scenarios for testing. (See col. 8, lines 31-34.) An
20 operator constructs the directed graph from a written body of requirements or from
21 some other requirements models or formalism. (See col. 10, lines 41-44.) The
22 Singh reference teaches that a user can select specific test cases in the regression
23 test case set to determine what affect, if any, the additional or new features have
24 on the remaining functionality. (See col. 13, lines 7-10.) The Singh reference is
25 silent on the testing of GUI applications. Rather, the Singh reference is directed at

1 testing procedural logic within a software application. In addition, the features
2 and the order for testing the features are determined at the time that the directed
3 graph is created.

4 *Traversal of Rejected Claims Based on the Cited References*

5 Amended **Claim 1** recites:

6 A method for testing, the method comprising:
7 retrieving information descriptive of *a graphics element rendered*
8 *during execution of the software being tested, the information identifying*
9 *an executable feature associated with the graphics element;*
10 *storing an association between the executable feature and the*
11 *graphics element in a map data structure containing information related*
12 *to at least one graphics element for testing, the association being stored in*
13 *the map data structure during execution of the software being tested;*
14 executing the executable feature associated with the graphics
15 element; and
16 updating the association in the map data structure upon execution of
17 the executable feature.

18 The present invention “provides a systematic approach to exploring
19 features of a GUI of a software application.” See page 12, lines 21-22, of the
20 present Application.) In general, the systematic approach is determined based on
21 information retrieved with respect to “a graphics element rendered during
22 execution of a software application being tested” as recited in independent Claims
23 1 and recited in similar manners for the other independent Claims 13 and 22.
24 Thus, the determination of what features are tested are determined automatically
25 during the testing of the target application. This is in direct contrast to the
teachings of the Parker reference which teaches that a script is written that
determines what features are tested and in what order the features are tested. This
is also in contrast to the teaching of the Singh reference which teaches to construct
a directed graph which determines what features and what order the features are

1 tested. Thus, the Examiner has not cited any reference that teaches or suggests the
2 claimed invention. In fact, even if all of these references could be combined, their
3 teachings could not possibly suggest the present invention. In addition, there is no
4 suggestion or motivation to combine these references.

5 Claim 1 further recites “storing an association between the executable
6 feature and the graphics element in a map data structure containing information
7 related to at least one graphics element for testing, the association being stored
8 in the map data structure during execution of the software being tested.” The
9 Parker reference does not describe such a process. The testing script according
10 to the Parker reference is not dynamic during the testing process. It is produced
11 before testing occurs; if changes are required, they must be made by a human
12 and while testing is not occurring. More specifically, the associations between
13 an executable feature and a graphics element are determined before a testing
14 process occurs. These associations are accounted for in the testing script. This
15 is explicitly shown in Table 2 of the Parker reference. In distinction, the claims
16 of the present invention recite that an association is stored “during execution of
17 the software being tested.” Thus, the testing process according to the instant
18 application has dynamic functionality. The other independent claims of the
19 instant application set forth similar subject matter as that found in Claim 1.

20 Parker teaches that the testing script can “identify GUI objects at the GUI
21 superclass level” without the use of GUI-specific names. (See column 16, lines
22 53-56.) The identification of GUI objects is done after the testing script was
23 created. Parker discloses that a test tool is used to “ask the GUI about an
24 object’s location and state just before making use of the information in order to
25 implement the test script’s request.” (See column 16 and 17, lines 64-67 and line

1 1, respectively.) Again modification of the testing script is not undertaken as
2 part of the requesting process. Parker states that the mapping “between high-
3 level logical object names and actual runtime GUI object names is one of the
4 most challenging tasks performed by the test tool. The test tool is not the testing
5 script and no modification of to testing script occurs as a result of the actions
6 taken by the test tool. Therefore, this particular disclosure of the Parker
7 reference also does not teach or suggest “storing an association between the
8 executable feature and the graphics element in a map data structure containing
9 information related to at least one graphics element for testing, the association
10 being stored in the map data structure during execution of the software being
11 tested.” (See Claim 1.) The other independent claims of the instant application
12 set forth similar subject matter as that found in Claim 1.

13 Thus, for at least one or more of the above reasons, the Applicant contends
14 that the Parker reference, whether considered alone or with any permissible
15 combination of prior art of record, does not teach or suggest each limitation recited
16 in independent Claims 1, 13, and 22. Therefore, the Applicant respectfully
17 submits that the §103 rejection of independent Claims 1, 13, and 22 is improper,
18 and respectfully requests reconsideration and withdrawal of this rejection.

19 Furthermore, the dependent claims include other limitations that are not
20 taught or suggested by the prior art of record. For example, “selecting executable
21 features in a depth-first mode of operation,” “selecting executable features in a
22 breadth-first mode of operation,” and other limitations are not taught or suggested.
23 Therefore, for at least the above reasons, Applicant respectfully submits that the
24 §103 rejection of dependent Claims 2-11, 14-15, 17-21, 23-28 is improper, and
25 respectfully requests reconsideration and withdrawal of this rejection.

1 *Legal Framework*

2 In overview, as stated in MPEP § 2143, to establish a prima facie case of
3 obviousness, three basic criteria must be met. First, there must be some
4 suggestion or motivation, either in the references themselves or in the knowledge
5 generally available to one of ordinary skill in the art, to modify the reference or to
6 combine reference teachings. Second, there must be a reasonable expectation of
7 success. Finally, the prior art reference (or references when combined) must teach
8 or suggest all the claim limitations. The teaching or suggestion to make the
9 claimed combination and the reasonable expectation of success must both be
10 found in the prior art, not in applicant's disclosure. In re Vaeck, 947 F.2d 488, 20
11 USPQ2d 1438 (Fed. Cir. 1991).

12 Further, as stated in MPEP § 2143.01, obviousness can only be established
13 by combining or modifying the teachings of the prior art to produce the claimed
14 invention where there is some teaching, suggestion, or motivation to do so found
15 either in the references themselves or in the knowledge generally available to one
16 of ordinary skill in the art. In re Fine, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir.
17 1988); In re Jones, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). The mere
18 fact that references can be combined or modified does not render the resultant
19 combination obvious unless the prior art also suggests the desirability of the
20 combination. In re Mills, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990).

21 Therefore, "all words in a claim must be considered in judging the
22 patentability of that claim against the prior art." In re Wilson, 424 F.2d 1382, 165
23 USPQ 494, 496 (CCPA 1970). If an independent claim is nonobvious under 35
24 U.S.C. 103, then any claim depending therefrom is nonobvious. In re Fine, 837
25 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).

1 With this legal framework in mind and for the additional reasons explained
2 above, the Applicant respectfully submits that the current rejection under 35
3 U.S.C. §103(a) is improper and must be withdrawn. **Conclusion**

4 Applicant has considered the other references cited by the Examiner in the
5 Office Action. None of these references appear to affect the patentability of
6 Applicant's claims. By the foregoing remarks, Applicant believes that pending
7 claims are allowable and the application is in condition for allowance. Therefore,
8 a Notice of Allowance is respectfully requested. Should the Examiner have any
9 further issues regarding this application, the Examiner is requested to contact the
10 undersigned attorney for the Applicant at the telephone number provided below.

11
12 Respectfully Submitted,

13
14 Dated: 5-9-2006

15 By: _____



16 Tim R. Wyckoff
17 Lee & Hayes, pllc
18 Reg. No. 46,175
19 (206) 315-4001 ext. 110
20
21
22
23
24
25